

Correspondance Python-Javascript

	Python	Javascript
Longueur d'un tableau, d'une chaîne de caractères	<i>len</i> (tableau)	tableau. <i>length</i>
Tableau vide	t=[]	<i>var</i> t= <i>new Array</i> ()
Ajout d'un élément dans un tableau	t. <i>append</i> (élément)	t. <i>push</i> (élément)
Appel d'un élément	t[i]	t[i]
Modulo	a%10	a%10
Division entière	a//10	<i>Math.trunc</i> (a/10)
Conversion nombre-> chaîne de caractères	<i>str</i> (nb)	nb. <i>toString</i> ()
Conversion chaîne de caractères -> nombre	<i>int</i> (chaîne) <i>float</i> (chaîne)	<i>parseInt</i> (chaîne) <i>parseFloat</i> (chaîne)
Conversion code ascii -> caractère	<i>chr</i> (entier)	<i>String.fromCharCode</i> (entier)
Conversion caractère -> code ascii	<i>ord</i> (caractère)	caractère. <i>charCodeAt</i> (0)
Obtenir le (i+1) ^{ème} caractère d'une chaîne	chaîne[i]	chaîne. <i>charAt</i> (i)
Conversion chaîne de caractères -> liste de caractères (tableau)	<i>list</i> (chaîne)	chaîne. <i>split</i> ("")

Structures algorithmiques

Python	Javascript
<i>for</i> i in range(1,len(b)) : instructions	<i>for</i> (var i =1; i<b.length; i++){ instructions }
<i>for</i> i in range(n,p) : instructions	<i>for</i> (var i =n; i<n; i++){ instructions }
<i>for</i> i in range(n,p,k) : # n<p et k>1 instructions	<i>for</i> (var i =n; i<n; i=i+k){ instructions }
<i>while</i> (test) : instructions	<i>while</i> (test) { instructions }
<i>if</i> (test): instructions <i>else</i> : instructions	<i>if</i> (test){ instructions } <i>else</i> { instructions }
<i>if</i> (expression==valeur1): instructions <i>elif</i> (expression==valeur2): instructions <i>elif</i> (expression==valeur3): . . . <i>else</i> : instructions	<i>switch</i> (expression) { <i>case</i> valeur1: // si expression=valeur1 instructions <i>break</i> ; <i>case</i> valeur2: instructions <i>break</i> ; ... <i>default</i> : instructions du else }